

(19)



(11)

EP 3 100 414 B1

(12)

EUROPEAN PATENT SPECIFICATION

(45) Date of publication and mention
of the grant of the patent:
26.09.2018 Bulletin 2018/39

(51) Int Cl.:
H04L 29/06 ^(2006.01) **H04L 12/26** ^(2006.01)
H04W 12/12 ^(2009.01) **H04W 40/02** ^(2009.01)

(21) Application number: **15702948.9**

(86) International application number:
PCT/US2015/012915

(22) Date of filing: **26.01.2015**

(87) International publication number:
WO 2015/116538 (06.08.2015 Gazette 2015/31)

(54) SESSION-BASED PACKET ROUTING FOR FACILITATING PACKET ANALYTICS

SITZUNGSBASIERTE PAKETWEITERLEITUNG ZUR VEREINFACHUNG DER PAKETANALYSE

ROUTAGE DE PAQUETS BASÉ SUR LA SESSION POUR FACILITER L'ANALYSE DE PAQUETS

(84) Designated Contracting States:
**AL AT BE BG CH CY CZ DE DK EE ES FI FR GB
GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO
PL PT RO RS SE SI SK SM TR**

(30) Priority: **28.01.2014 US 201461932650 P**
16.05.2014 US 201461994693 P
05.12.2014 US 201462088434 P
22.01.2015 US 201514603304

(43) Date of publication of application:
07.12.2016 Bulletin 2016/49

(73) Proprietor: **Extreme Networks, Inc.**
San Jose, CA 95119 (US)

(72) Inventors:
• **HSU, Ivy Pei-Shan**
Dublin, California 94568 (US)

- **CHHABRIA, Sanjeev Nand**
Castro Valley, California 94552 (US)
- **CHEN, Xiaochu**
San Ramon, California 94582 (US)
- **MUNSHI, Sanjay**
San Ramon, California 94582 (US)
- **NARASIMHAN, Arvindsrinivasan Lakshmi**
San Jose, California 95134 (US)

(74) Representative: **Müller-Boré & Partner**
Patentanwälte PartG mbB
Friedenheimer Brücke 21
80639 München (DE)

(56) References cited:
EP-A1- 2 654 340 WO-A1-2010/135474
IE-A1- 20 070 438 US-A1- 2012 157 088

Note: Within nine months of the publication of the mention of the grant of the European patent in the European Patent Bulletin, any person may give notice to the European Patent Office of opposition to that patent, in accordance with the Implementing Regulations. Notice of opposition shall not be deemed to have been filed until the opposition fee has been paid. (Art. 99(1) European Patent Convention).

EP 3 100 414 B1

Description

BACKGROUND

[0001] The disclosure herein pertains generally to the field of telecommunications and more specifically to techniques for routing duplicates of data packets to analytic servers.

[0002] General Packet Radio Service (GPRS) is a standard for wireless communications which enables data to be transmitted at speeds up to 115 kilobits per second, compared with Global System for Mobile Communications (GSM) systems' 9.6 kilobits per second. GPRS, which supports a wide range of bandwidths makes efficient use of limited bandwidth and is suitable for sending and receiving small bursts of data, such as e-mail and Web browsing, as well as large volumes of data.

[0003] GPRS Tunneling Protocol (GTP) is a group of Internet Protocol (IP)-based communications protocols used to carry packets conforming to the GPRS standard within GSM, UMTS and LTE networks. In Third Generation Partnership Project architectures, GTP and Proxy Mobile IPv6-based interfaces are specified on various interface points. GTP can be decomposed into separate protocols, including GTP-C and GTP-U. In 3G and 4G wireless networks, GTP-C messages are control messages used between the network elements to activate and de-activate sessions originating from mobile user endpoints. As an example, in 3G networks, GTP-C is used within a GPRS core network for signaling between gateway GPRS support nodes (GGSN) and serving GPRS support nodes (SGSN). This allows the SGSN to activate a session on a user's behalf, to deactivate the same session, to adjust quality of service parameters, or to update a session for a subscriber who has just arrived from another SGSN. GTP-U is used for carrying user data within a GPRS core network and between a radio access network and the core network. The user data transported can be packets in any of IPv4, IPv6, or Point-to-Point Protocol (PPP) formats.

[0004] An operator of a telecommunication network can find it beneficial to analyze the traffic that flows through that network. Such analysis might be performed for a variety of different reasons. For example, the operator might want to obtain information that could be used as business intelligence. For another example, the operator might want to detect and pre-empt attacks being made through the network. In order to help prevent such attacks, the operator might want to analyze traffic to determine the sources from which different types of traffic originate.

[0005] Such traffic analysis can be performed at an analytic server that the operator maintains. Data packets flowing through the network can be duplicated, and the duplicate packets can be diverted to such an analytic server. Due to the vast amount of traffic that flows through the network, the operator might maintain numerous separate analytic servers that are capable of analyzing dif-

ferent portions of the total traffic concurrently.

[0006] The traffic flowing through a telecommunications network often will represent multiple separate and distinct communication sessions. Such sessions can originate from different mobile devices. Regarding GPRS, a session is a tunnel that is established between two endpoints in a communication network. Communications between those endpoints passes through this established tunnel. In a 3G network, the session is established through the creation of a packet data protocol (PDP) context—a data structure—on both an SGSN endpoint and a GGSN endpoint. This data structure contains session information, contents of which are described further below. The establishment of the session allocates a PDP context in the SGSN with which the mobile device is currently in communication. The establishment of the session further allocates that PDP context in the GGSN that serving the mobile device user's access point. The data recorded in the PDP context includes: the mobile device's Internet Protocol (IP) address, the mobile device's International Mobile Subscriber identity (IMSI), a Tunnel Endpoint ID (TEID) at the GGSN, and a Tunnel Endpoint ID (TEID) at the SGSN.

[0007] IE 2007/0438 discloses a monitoring system comprising a data collector for passively reading content being communicated in a network (SGSN-GGSN) and a processor which filters the content to identify request messages and associated subscribers. The system parses URLs of downloaded content into individual components to recognize the content, opens a record in a session cache for the request message and the content, and writes a flag to the record to indicate that the content is routed to the requesting subscriber. Further, the system parses the content to identify key words indicating the content, and generates an output indicating quality of the content response to the request message.

[0008] As is mentioned above, traffic analysis can be performed at an analytic server that an operator maintains. In order for the analysis to be complete, it is desirable that all traffic belonging to a particular communication session be diverted to the same analytic server under circumstances in which multiple analytic servers are analyzing the network traffic.

[0009] Achieving this result can be difficult due to the fact that the traffic in a mobile telecommunications network can originate from mobile devices that, by their nature, tend to move about geographically. As a mobile device moves from one region to another, the mobile device may leave the range of one cellular telephone tower and come into the range of another cellular telephone tower. The point through which the mobile device accesses the telecommunication network can thus change, as the device moves, from one point (e.g., a first cellular telephone tower) to another point (e.g., a second cellular telephone tower). When such a change occurs, the parameters associated with the mobile device's currently active communication sessions are likely to change as well.

[0010] The change in these communication session parameters complicates the task of ensuring that, for each communication session, all of that communication session's traffic will be sent to the same analytic server in a group of such servers. If the parameters associated with a particular communication session change due to the mobile device moving, then network elements that select the analytic server to which duplicate packets should be forwarded might accidentally send subsequent traffic belonging to that particular communication session to a different analytic server than the one to which those network elements had been sending that traffic prior to the change.

SUMMARY

[0011] The invention relates to a network node as defined in the claims. In certain embodiments, a OTP correlation cluster (GCC) can automatically program a network element, such as a switch, to forward copies of packets originating from a mobile device and having a shared attribute to the same port of that network element. The network element thereby sends the packet copies having the shared attribute to the same analytic server, regardless of the regions into which the mobile device moves.

[0012] To accomplish this, the GCC can monitor attributes of copies of control packets that the network element receives. For example, the GCC can observe source and destination attributes within control packet copies. In response to detecting a changed source attribute (e.g., SGSN) within a control packet originating from a mobile device, the GCC can update a session map specific to that mobile device in order to cause packets having that changed source attribute to be forwarded to the same port to which packets having the former source attribute were being forwarded prior to the change. As a result, the network element can ensure that packets belonging to a particular session still are forwarded to the same analytic server even if the mobile device has moved to a different region.

[0013] When the network element receives a copy of a control packet originating from a mobile device, the network element can forward that control packet copy to the GCC. Information within the control packet copy uniquely identifies the mobile device. The GCC can maintain a separate session map for each separate mobile device.

[0014] Using attributes contained in the control packet, the GCC can create a new session map specifically for the mobile device if one does not already exist. The new session map can associate an identifier of a particular port of the network element with a combination of source and destination attributes (e.g., SGSN and GGSN) contained within the control packet. The GCC can choose the particular port by inputting the combination of source and destination attribute values into a specified function.

[0015] By determining whether the output of the spec-

ified function produces a same port identifier as the one specified within an existing session map for the mobile device, the GCC can determine whether a part of that session map has become invalid. Such invalidity can result from the mobile device moving from one area to another, consequently producing a control packet having the same destination attribute (e.g., GGSN) but a different source attribute (e.g., SGSN).

[0016] In response to determining that a part of the session map has become invalid, the GCC can update its session map for the mobile device by changing the source attribute in that part of the session map. Such changing can involve replacing that part of the session map's former source attribute with the control packet's source attribute, while retaining the port that was already specified in that part of the session map. In conjunction with updating the session map, the GCC can program forwarding rules (e.g., an access control list) within the network element. The network element follows these forwarding rules to forward packet copies having this updated attribute combination to the same port to which packet copies having the previous attribute combination were previously forwarded.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017]

FIG. 1 is a system diagram illustrating an example of a mobile telecommunication network including a GTP correlation cluster (GCC), according to some embodiments.

FIG. 2 is a system diagram illustrating an example of a system including a network switch that includes a GTP services card that forwards control packets to a GCC, according to some embodiments.

FIG. 3A is a flow diagram illustrating a parts of a technique that a network switch can perform relative to duplicate GTP packets in order to ensure that such packets belonging to a particular communication session will continue to be directed to the same analytic server regardless of the sub-network from which those packets arrive at the network switch, according some embodiments.

FIG. 3B is a flow diagram illustrating parts of a technique that a network switch can perform specifically relative to duplicate GTP-C packets in order to determine whether a communication session has moved to a different sub-network, and to program a network switch's GCL accordingly to ensure consistent forwarding of the communication session's duplicate packets to a particular analytic server, according to some embodiments.

FIG. 4 is a system diagram illustrating an example

of a system including a GCC that stores multiple different session maps for multiple different communication sessions, according to some embodiments.

FIG. 5 is a diagram illustrating an example of a populated session map, according to some embodiments.

FIG. 6 depicts a simplified block diagram of a network device that may incorporate some embodiments.

FIG. 7 is a flow diagram illustrating an overview of a technique by which a network node, such as a network switch, can forward packets belonging to a same session to a same analytic server regarding of the diverse network entry points at which those packets entered a telecommunication network, according to some embodiments.

FIG. 8 is a system diagram illustrating an example of a network switch that contains a GCC and analytic modules, according to some embodiments.

FIG. 9 is a diagram illustrating another example of a populated session map, according to some embodiments.

DETAILED DESCRIPTION

[0018] In the following description, for the purposes of explanation, specific details are set forth in order to provide a thorough understanding of embodiments of the invention. However, it will be apparent that various embodiments may be practiced without these specific details. The figures and description are not intended to be restrictive.

[0019] Techniques disclosed herein can be used to ensure that a network element, such as a network switch, continues to forward copies of packets originating from a mobile device and having a common attribute (e.g., destination attribute) value to a same port of that network element, thereby to be sent to the same analytic server, regardless of the regions into which the mobile device moves. These techniques can be used to ensure that the network element will not cause copies of packets originating from a mobile device and having a common attribute value to be forwarded to different analytic servers as a result of the mobile device interfacing with different network entry points as the mobile device moves from region to region.

[0020] FIG. 7 is a flow diagram illustrating an overview of a technique by which a network node, such as a network switch, can forward packets belonging to a same session to a same analytic server regarding of the diverse network entry points at which those packets entered a telecommunication network, according to some embodiments. Although FIG. 7 illustrates certain operations being performed in a certain order, some embodiments can

involve additional, fewer, or different operations being performed in potentially different orders.

[0021] In block 702, a network node receives a first packet from a first network entry point. For example, a network switch can receive the first packet via a first cellular telephone tower through which a mobile device was interfacing with a telecommunications network when the mobile device originated the first packet.

[0022] In block 704, the network node receives a second packet from a second network entry point. Continuing the example, the network switch can receive the second packet via a second cellular telephone tower through which the mobile device was interfacing with the telecommunications network when the mobile device originated the second packet. The second cellular telephone tower may be in a different location than the first cellular telephone tower. In between the times that the mobile device originated the first and second packets, the mobile device might have moved out of communication range of the first cellular telephone tower and into communication range of the second cellular telephone tower.

[0023] In block 706, the network node determines whether the second packet belongs to a same session to which the first packet belongs. Continuing the example, the network switch can make this determination based on a session mapping between an identifier of the mobile device and attributes of the second packet. Techniques for generating this mapping are described in greater detail below.

[0024] In block 708, in response to determining that the second packet belongs to the same session to which the first packet belongs, the network node forwards the second packet to an analytic server to which the first packet was forwarded. Continuing the example, based on the session mapping, the network switch can determine that the second packet's attributes are mapped to a same identifier to which the first packet's attributes were formerly mapped, and can consequently conclude that the second packet belongs to the same session to which the first packet belongs. Consequently, the network switch can forward the second packet to a same analytic server to which the network switch previously forwarded the first packet, thereby ensuring that packets from the same session will not be diverted to different analytic servers even if the originating mobile device has accessed the telecommunications network from different network entry points.

DIVERTING DUPLICATE GTP TRAFFIC TO ANALYTIC SERVERS

[0025] In a telecommunications network, a network switch can receive packets that originated from different mobile devices and flowed through different cellular towers and network entry points. Such packets can belong to different sessions, with each session identified by a combination of attributes (e.g., an IMSI and a destination attribute value). The network switch can forward copies

of these packets to various separate analytic servers. The network switch can also forward the original packets on toward their intended destinations. Using values of attributes specified within the packet copies, the network switch can choose the ports of the network switch to which the packet copies are to be forwarded. Some embodiments described herein ensure that copies of packets belonging to a particular session continue to be forwarded to the same port, and therefore analytic server, even if a value of some attribute of the packets (e.g., a source attribute value) changes. In some embodiments described below, a network switch can use a GTP correlation cluster (GCC) in order to achieve this result.

[0026] FIG. 1 is a system diagram illustrating an example of a mobile telecommunication network 100 including a GTP correlation cluster (GCC), according to some embodiments. A GCC can be a device or program that maintains separate session maps for separate mobile devices, that uses information contained in copies of control packets to update the session maps, and that automatically programs data structures in a network element based on the session maps in order to cause the network element to forward copies of packets to analytic servers in a consistent manner. Network 100 includes both a 3G network and a 4G/LTE network. The 3G network includes a mobile device 102, a support GPRS support node (SGSN) 106, and a gateway GPRS support node (GGSN) 108. The 4G/LTE network includes an mobile device 110, an Evolved Node B (eNodeB) 112, a mobile management entity (MME) 113, a serving gateway (SGW) 114, and a packet data network gateway (PGW) 116. Traffic from both networks flows into a network switch 118. An example of network switch 118 is the MLXe L2/L3 switch from Brocade Communications Systems, Inc. of San Jose. Network switch 118 interfaces with a GTP correlation cluster (GCC) 120. Network switch 118 also stores a set of flow rules 132.

[0027] A tunnel through the network, defined by values within data structures established at network endpoints during a communication session's creation, can be generated for each communication session. Each communication session can be identified by a combination of values of attributes contained within a control packet that is transmitted to establish the communication session (such control packets also can be transmitted when a mobile device begins to interface with a different network entry point (e.g., cell tower)). In some embodiments, each time that a mobile device interfaces with a different cellular tower or other wireless network access point, the mobile device transmits a new control packet identifying a source corresponding to that wireless network access point. The attributes whose values define a session can include, for example, an IMSI and a destination attribute (e.g., GGSN). Packets that travel through a tunnel generated for a session belong to that session.

[0028] The communication sessions to which different subsets of traffic entering network switch 118 belong can be identified using GTP correlation cluster (GCC) 120.

Based on these identified communication sessions, network switch 118 forwards copies of packets belonging to the subsets through its ports on toward the various ones of analytic servers 130A-N that are performing analysis relative to the traffic associated with those sessions. Network switch 118 forwards the original packets on towards their intended destinations. Each of analytic servers 130A-N can be associated with a different set of communication sessions. Each of analytic servers 130A-N can perform analysis relative to the packet copies belonging for its associated communication sessions. As a result, no communication session is divided among separate ones of analytic servers 130A-N. In one embodiment, there are 48 separate analytic servers 130A-N, each connected to a separate port of network switch 118. Each of analytic servers 130A-N can receive packet copies, classify those packet copies into different types based on rules and packet attribute values, and determine network sources from which different types of traffic originate. In this manner, for example, network traffic that appears to be associated with an attack, such as a denial-of-service attack, can be identified, and the sources of the attack can be ascertained.

[0029] In some telecommunications networks (one example being the 3G network depicted in FIG. 1), traffic originating from mobile devices such as mobile device 102 can flow from the mobile device to cell phone tower 104 to SGSN 106 to GGSN 108 to network switch 118. Network switch 118 can forward this original traffic on to its intended destinations. Additionally, a tap can be inserted on the connection between SGSN 106 and GGSN 108. A part of the traffic signal flowing through that connection is thereby duplicated and diverted to network switch 118 for analytic purposes. This duplicate data includes GTP-C (control packets) and GTP-U (data packets) 126. Traffic flowing back from network switch 118 toward the mobile devices passes between GGSN 108 to SGSN 106 in the opposite direction. The tap on the connection therefore additionally duplicates a part of the traffic signal flowing back through that connection and also diverts that traffic signal to network switch 118 for analytic purposes.

[0030] The traffic that network switch 118 receives can include packets from multiple different communication sessions. Multiple communication sessions, each involving packets belonging to a particular one of those sessions, can be assigned to a particular one of analytic servers 130A-N. The several communication sessions that may be assigned to a particular analytic server constitute a subset of the total traffic that network switch 118 receives. Each of analytic servers 130A-N can be assigned a different subset of the total traffic, with each subset including multiple communication sessions. Potentially through consultation with GCC 120 (possibly through the application of a hash algorithm), network switch 118 can identify a communication session to which a subset of duplicated traffic, and its constituent packets, belongs. For each duplicated packet, network switch 118

can select, from its ports, the particular port that is associated with the communication session to which that packet belongs. Network switch 118 can then forward the duplicate packet through that selected port to the proper one of analytic servers 130A-N that is handling analytics for that particular communication session.

[0031] In some telecommunications networks (one example being the 4G network depicted in FIG. 1), traffic originating from mobile devices can flow from mobile device 110 to eNodeB 112 to serving gateway (SGW) 114 to packet data network gateway (PGW) 116 to network switch 118. Traffic also can flow from eNodeB 112 to MME 113 to SGW 114. Network switch 118 can forward this original traffic on to its intended destinations. Additionally, a duplication mechanism along the path from mobile device 110 to network switch 118 can generate duplicates of the packets flowing along that path. One example of such a duplication mechanism is a tap that can be inserted on the connection between SGW 114 and PGW 116. Packets flowing through that connection arc thereby duplicated and diverted to network switch 118 for analytic purposes. These duplicate packets may include GTP-C (control packets) and GTP-U (data packets) data 128. Packets flowing back toward the mobile devices passes between SGW 114 to mobile device 110 in the opposite direction. The duplication mechanism (e.g., a tap on the connection) therefore additionally duplicates packets flowing back through that connection and also diverts those duplicated packets to network switch 118 for analytic purposes. Additionally or alternatively, duplication mechanisms can include taps elsewhere along paths between mobile device 110 and network switch 118. For example, a tap can be inserted on a connection between eNodeB 112 and SGW 114. GTP-U packets 127 flowing through that connection are thereby duplicated and diverted to network switch 118 for analytic purposes. For another example, a tap can be inserted on a connection between MME 113 and SGW 114. GTP-C packets 125 flowing through that connection are thereby duplicated and diverted to network switch 118 for analytic purposes.

[0032] Again, through consultation with GCC 120, network switch 118 can identify a communication session to which a duplicate packet belongs. For each duplicate packet, network switch 118 can select, from its ports, the particular port that is associated with that packet's communication session. Network switch 118 can then forward the duplicate packet through that port to the proper one of analytic servers 130A-N that is handling all duplicate data packets for the communication session to which those packets belong.

[0033] If a particular mobile device is executing an application that is associated with a particular communication session, then during the course of that particular communication session the mobile device may travel out of range of one cellular telephone tower and into the range of another cellular telephone tower. This change in access point can cause at least some of the parameters

(e.g., source) of the particular communication session to change. However, using techniques described herein, in spite of this change, network switch 118 can continue to send, to the same one of analytic servers 130A-N, the duplicated traffic for that communication session both before and after the movement.

[0034] Each mobile device is typically associated with a unique International Mobile Subscriber Identity (IMSI) number. When a mobile device establishes a new communication session, the mobile device includes its unique IMSI within the first control packet that the mobile device sends through the telecommunication network. Each communication session can be identified based on a combination of attributes, such as IMSI and destination address, for example. The control packet indicates the source IP address from which subsequent non-control data packets belonging to the same communication session as the control packet will originate. However, although those subsequent non-control data packets typically do indicate the source IP address, they typically do not indicate the IMSI number. Some embodiments may use an International Mobile Station Equipment Identity (IMEI) of a mobile device in conjunction with or in place of the IMSI as described herein.

[0035] Due to potential changes in source IP address caused by a mobile device travelling from region to region as discussed above, the non-control data packets belonging to a particular communication session may or may not specify all the same IP addresses that the control packet establishing that session specifies. The data packets might not follow the same path through the network that the control packet follows. While a mobile device is communicating with the mobile telecommunication network through a first cell tower, the data packets from that mobile device may indicate a first source IP address (e.g., for an SGSN) and destination IP address (e.g., for a GGSN) pair, but later, while the same mobile device is communicating with the mobile telecommunication network through a second cell tower during the same session, the data packets now originating from that same mobile device may indicate a second source IP address (e.g., for a different SGSN) and destination IP address (e.g., for the same GGSN) pair different from the first pair. These non-control data packets typically do not specify the IMSI number that was specified by the control packet for the same session.

[0036] After a communication session has been established, the data packets belonging to that communication session start to flow through the network. As is discussed above, the movement of a mobile device from one region to another, causing the mobile device to interface with the telecommunication network through a different network entry point, could cause the entry point earlier specified in one of a session's data packets to differ from an entry point later specified in a subsequent data packet belonging to the same session. Network switch 118 is tasked with the responsibility of ensuring that regardless of the entry point of the duplicated data packets belonging

to a particular communication session, and regardless of the port of network switch 118 on which those duplicated data packets arrive, all of those duplicated data packets belonging to the particular communication session will be forwarded to the same one of analytic servers 130A-N. Network switch 118 cannot rely upon the duplicate non-control data packets it receives to supply the IMSI number of the non-control packets' origin, because that number is typically not present within those non-control packets; the IMSI number is typically specified within control data packets transmitted during a session's establishment, prior to the transmission of the non-control data packets belonging to that session.

[0037] In some embodiments, network switch 118 can ensure that non-control data packets become associated with the proper one of analytic servers 130A-N in spite of possible differences in some of the parameters (e.g., source IP address) specified within those non-control data packets. Network switch 118 can use the services of GCC 120 to determine that a packet is to be forwarded to a specific analytic server regardless of whether some of that packet's parameters differ from parameters of other packets belonging to the same communication session. In certain embodiments, GCC 120 can observe control packets that a mobile device transmits whenever the mobile device begins to interface with a different network entry point (e.g., cell tower). GCC 120 can update a mapping between the IMSI specified in those control packets and the new parameter value(s) contained within those control packets. GCC 120 can use this updated mapping to program network switch 118 to forward non-control packets having those new parameter value(s) to the same analytic server to which non-control packets having the previous parameter value(s) were previously forwarded. GCC 120 can execute on a device separate from network switch 118, or within network switch 118 itself. Through such programming, GCC 120 can update, in network switch 118, forwarding rules 124 (e.g., a GCL) that network switch 118 follows in selecting the ports through which duplicate traffic is to be forwarded to various ones of analytic servers 130A-N. Network switch 118 sends GTP-C (control packet) 122 to GCC 120.

CONSISTENT ROUTING BY COMPENSATING FOR MOBILE DEVICE MOVEMENT

[0038] The source and destination IP addresses specified in one control packet belonging to a particular communication session might be the same as or different from the source and destination IP addresses specified in a subsequent control packet belonging to the same particular communication session due, for example, to the source IP address changing when a mobile device moves to a different region, as described above. When the address pairs of the control packets match, this is indicative of the situation in which those packets are entering the network through the same cellular telephone tower, and therefore passing through the same SGSN-

GGSN pair (in the case of a 3G network) or the same MME-SGW pair (in the case of a 4G/LTE network). Discussed below are systems and techniques for ensuring a network switch will forward copies of non-control packets belonging to a particular communication session to the same analytic server even if address pairs of non-control packets within that particular communication session differ due to mobile device movement. As is described further in greater detail, a network switch can use the services of a GCC in order to ensure this consistent forwarding behavior. A system that includes a GCC is described below in connection with FIG. 2.

[0039] FIG. 2 is a system diagram illustrating an example of a system 200 including a network switch that includes a GPRS Tunneling Protocol (GTP) services card that forwards mirrored control packets to a GCC, according to some embodiments. Network switch 118, and more specifically a line card (e.g., a 10G line card) 206, can be configured with configuration data that indicates that packets received through a particular ingress port X of switch 118 (and card 206) are to be internally routed to a particular service card (e.g., line card 208) of switch 118. Card 206 can receive packets through a network 202 which can include various sub-networks including the Internet or portions thereof. In FIG. 2, the particular service card to which card 206 is configured to forward the designated packets is line card 208 of switch 118. Card 208 can be configured to act as a dedicated GTP service card instead of sending packets out from or receiving packets into switch 118. Line card 208, in turn, can internally route certain packets to a GCC, as will be discussed in greater detail below.

[0040] The configuration stored on card 206 might specify that packets received on any of ingress ports 1-4 are to be internally routed to a particular slot in which card 208 is situated. Under such circumstances, ingress ports 1-4 would be the ports through which network switch 118 is configured to receive duplicated traffic; these ingress ports would be the ports to which the duplication mechanisms (e.g., taps) described above interface, either through network cables or through wireless communications. As a result, duplicated traffic diverted from the duplication mechanisms (e.g., taps) is routed within network switch 118 to a dedicated GTP service card-in this example, card 208-specified in the configuration data. The dedicated GTP services card 208 is configured to receive packets from various line cards and to send packets to other line cards within network L2/L3 switch 118.

[0041] The dedicated GTP services card-in this example, card 208-can be configured to store data that designates each of the egress ports of network switch 118 that are connected to analytic servers 130A-N. The configuration data can be stored on card 208. These egress ports-also called telemetry ports-can be specified by a slot-and-port combination, for example. For example, assuming that there are 48 analytic servers 130A-N, ports 15/1 through 15/24 (or, ports 1-24 of a line card in slot

15 of network switch 118) and ports 16/1 through 16/24 (or, ports 1-24 of a line card in slot 16 of network switch 118) may be specified, in the configuration of the dedicated GTP services card (which might be in slot 10), as being the egress ports, or telemetry ports, that are connected to analytic servers 130A-N. Although ports are described herein in terms of being "ingress" or "egress" ports, such ports are not necessarily dedicated to only sending or receiving packets; rather, herein, a port is described as being an ingress port when it is to be used for that purpose relative to a data packet, and a port is described as being an egress port when it is to be used for that purpose relative to a data packet. A particular port can alternatively serve as an ingress port or an egress port relative to different packets. In FIG. 2, the line cards that include these egress ports are shown as line card 212 (e.g., in slot 15), and line card 214 (e.g., in slot 16).

[0042] Within network switch 118, original packets can be routed internally towards their intended destinations, while duplicate packets (known to be duplicates due to the ports on which they were received) can be routed internally based on whether they are control packets or non-control packets. Duplicate packets (both control and non-control) can be routed internally toward appropriate ones of analytic servers 130A-N, while duplicate control packets (or mirrors of those duplicate control packets) specifically also can be routed internally toward a GCC. The internal routing of a duplicate control packet can proceed from a line card containing the ingress port on which that packet was received, to the dedicated GTP services card, to another line card that forwards the packet to a GCC. In some embodiments, the internal routing of a duplicate control packet may involve other paths within switch 118 to reach a GCC. The discussion that follows pertains specifically to the internal routing of duplicate control packets. By way of an example, the dedicated GTP services card-in this example, card 208-can be configured to store data that instructs that duplicate control packets (i.e., GTP-C packets) received by the dedicated GTP services card are to be internally forwarded to a specified ingress port of a line card in a specified slot of network switch 118. For example, the dedicated GTP service card might be configured to mirror control packets to port 13/1 (or, port 1 of a line card in slot 13 of network switch 118). The line card that receives the duplicate control packets sends the duplicate control packets on to GCC 120. In FIG. 2, this line card is shown as line card 210. (e.g., in slot 13).

[0043] Based on the information that it receives over time, GCC 120 can create and maintain a session map 204. The purpose and use of session map 204 is discussed further below. Generally, session map 204 can be used to ensure that data packets belonging to a particular communication session will continue to be directed to the same one of analytic servers 130A-N regardless of the sub-network from which those data packets arrive at network switch 118.

[0044] Line card 208, which functions as the dedicated

GTP services card, can store a special type of access control list (ACL) called a GTP control list (GCL), shown in FIG. 2 as GCL 216. GCL 216 is an ACL specially customized for GTP traffic. In some embodiments, the forwarding rules discussed above are defined within GCL 216. Packets arriving at line card 208 possess designated attributes which may or may not be currently reflected within any GCL entry. If a particular packet's designated attributes are currently included in a GCL entry, then that particular packet is deemed to be a "hit" relative to GCL 216. Conversely, if the particular packet's designated attributes are not currently included in any GCL entry, then that particular packet is deemed to be a "miss" relative to GCL 216. If a particular packet is a "miss" relative to GCL 216, it means that GCL 216 has not yet been programmed with an entry that indicates the egress port to which packets having the particular packet's designated attribute values are to be forwarded. Such a situation could arise due to the particular packet belonging to a new session, or due to one or more of the values of the particular packet's designated attributes being different from those of other packets belonging to the same session (e.g., due to the origin mobile device having moved to a different region as discussed above). Over time, GCL 216 can be updated and augmented to include additional entries. GCC 120 can update GCL 216 through programming, by automatically adding or modifying entries within GCL 216. Entries in GCL 216 can map packet attribute values (e.g., values for source and destination IP addresses) to specific egress line cards and specific egress ports of those line cards. A technique by which GCC 120 determines how to modify entries within GCL 216 is discussed in greater detail below.

35 GTP CORRELATION CLUSTERING TECHNIQUE

[0045] As is discussed above, a GCC can automatically add or modify entries within a GCL, which a network switch uses to determine the egress ports to which duplicate non-control packets containing various attribute values are to be forwarded. By automatically programming the GCL in this manner, the GCC ensures that packets belonging to a particular session will continue to be routed toward the same analytic server to which previous packets belonging to that session were routed. The GCC can change GCL entries associated with a particular egress port of the network switch to reflect the new packet attribute value(s) (e.g., source/GGSN IP address) resulting from an originating mobile device's movement to a different region (and into a different sub-network), so that the network switch will route packets having those new attribute values toward that same egress port. FIG. 3A is a flow diagram illustrating parts of a technique that a network switch can perform relative to duplicate GTP packets in order to ensure that such packets belonging to a particular communication session will continue to be directed to the same analytic server regardless of the sub-network from which those packets arrive at the net-

work switch, according to some embodiments. The technique illustrated in FIG. 3A can be performed by network switch 118, for example. Related parts of a technique that network switch 118 can perform relative to duplicate GTP-C (control) packets specifically are discussed further below in connection with FIG. 3B.

[0046] In block 302, an ingress line card of the switch can receive a duplicate GTP packet, which might be a GTP-C packet or a GTP-U packet. The packet may be known to be a duplicate due to the packet having arrived on an ingress port on which only duplicate packets can arrive (e.g., a port communicatively coupled to a duplication mechanism such as a tap). Although some embodiments apply specifically to GTP packets and the GTP protocol, techniques described herein can be adapted to packets that conform to protocols other than GTP. For example, line card 206 can receive this packet through network 202. In block 304, the ingress line card can remove a multiprotocol label switching (MPLS) label (if present) from the received packet. For example, line card 206 can strip off this MPLS label. In block 306, the ingress line card can forward the processed packet to a GTP services card. For example, line card 206 can forward the packet, from which the MPLS label has been removed, to line card 208, which is the designated GTP services card.

[0047] In block 316, the GTP services card can determine whether the packet is a GTP-C (control) packet. For example, line card 208 can make this determination based on information, such as a transmission control protocol (TCP) port number, contained within the packet. If the packet is a GTP-C packet, then, in block 308, the GTP services card can send a copy of the GTP-C packet toward the GCC. Subsequent operations performed relative to the copy of the GTP-C packet are shown in FIG. 3B. Following the operations of block 308, or if the packet is not a GTP-C packet, control passes to block 310.

[0048] In some embodiments, the GCC stores GCL rules that load-balance certain GTP packet attribute ranges (e.g., SGW or SGSN IP address ranges) to specific analytic servers. In some embodiments, the GCC stores dynamic GCL rules to ensure that mobile device movement will not cause a change the analytic server to which GTP packets belonging to a particular session are forwarded. The GCC can program the network switch with both default and dynamic GCL rules.

[0049] In block 310, the GTP services card can attempt to match attributes of the packet to dynamic GCL rules, which are installed by the GCC, to determine an analytics port through which the packet is to be forwarded. In block 311, the GTP services card can determine whether the packet's attributes match any of the dynamic GCL rules. If the packet's attributes match a dynamic GCL rule, then control passes to block 314. Otherwise, control passes to block 312.

[0050] In block 312, in the event that there is no dynamic GCL rule match, the GTP services card can match default GCL rules, installed by the GCC (like the dynamic

GCL rules), to determine an analytics port through which the packet is to be forwarded. The GTP services card can select a particular line card's port that is mapped to or identified by an default GCL rule for forwarding out of the network switch. For example, line card 208 can select a port on line card 212 or 10G line card 214, depending on the determined analytics port. That port can be connected to a corresponding one of analytic servers 130A-N. Duplicate packets having a particular combination of source and destination IP addresses can be directed to the same port and therefore to the same one of analytic servers 103A-N, though packets having different source and destination IP addresses can be directed to different ports of line cards 212-214 and therefore to different ones of analytic servers 103A-N. Control then passes to block 314.

[0051] In block 314, the GTP services card can forward the packet to a particular line card's port that is mapped to or identified by a matching dynamic or default GCL rule. For example, line card 208 can forward the particular packet to line card 212 or line card 214, depending on the information contained in the matched rule. That line card can forward the particular packet out through the rule-designated port that is connected to a corresponding one of analytic servers 130A-N. The technique illustrated in FIG. 3A concludes relative to the particular packet, although the technique can be performed again relative to other duplicate GTP packets.

[0052] In some cases, an analytic server identifier, associated with one or more analytics ports, can be determined based on a combination of the particular packet's source IP address and destination IP address. The GTP services card can obtain these addresses from the particular packet's outer header. For example, using a specified function, line card 208 can translate these addresses into the analytic server identifier. The specified function may include a hash algorithm, for example. Conceptually, the application of the specified function to the combination of addresses distributes communication sessions among different analytic servers, such that combinations that produce the same analytic server identifier become associated with the same analytic server. This results in duplicate packets having the same addresses being routed to the same analytic server, so that as long as those addresses do not change between packets belonging to the same communication session, all of that communication session's duplicate packets will be routed to the same analytic server. In some embodiments, the combination can include additional or alternative data. In some embodiments, the specified function is designed to produce the same output regardless of whether the source IP address is swapped with the destination IP address in the input. In some embodiments, the analytic server identifier indicates a line card's port (e.g., card and port combination) through which duplicate packets can be forwarded out of the network switch toward the associated analytic server.

[0053] FIG. 3B is a flow diagram illustrating parts of a

technique that a network switch can perform specifically relative to duplicate GTP-C packets in order to determine whether a communication session has moved to a different sub-network, and to program a network switch's GCL accordingly to ensure consistent forwarding of the communication session's duplicate packets to a particular analytic server, according to some embodiments. Upon determining in block 316 of FIG. 3A that the packet is a GTP-C packet, in block 318 of FIG. 3B, the GTP services card can forward a copy of the particular GTP-C packet to a designated line card that interfaces with the GCC. For example, line card 208 can forward a particular GTP-C packet to line card 210. Inasmuch as some embodiments may involve duplicate GTP-C packets being sent to both an analytic server (for analysis along with all of a communication session's other duplicate packets) and the GCC, some embodiments can involve creating a further copy, or mirror, of each duplicate GTP-C packet specifically for transmission to the GCC. It can be such a mirror of the GTP-C packet the GTP services card transmits to the GCC. In block 320, the designated line card that interfaces with the GCC can send the particular GTP-C packet to the GCC. For example, line card 210 can send the particular GTP-C packet to GCC 120. In some embodiments, this sending can involve forwarding a copy of the particular GTP-C packet out of a port of line card 210. In some embodiments, this sending can involve passing a copy of the particular GTP-C packet to an input data interface of GCC 120. Control passes to block 330.

[0054] In block 330, the GCC can determine whether a session map for a mobile device from which the particular GTP-C packet originated exists by determining whether an identifier contained in the particular GTP-C packet matches an identifier of an existing session map maintained by the GCC. For example, assuming that session map 204 already exists, GCC 120 can determine whether an IMSI (which identifies a mobile device) specified in the particular GTP-C packet matches an IMSI specified within session map 204. If a session map for the mobile device exists, as evidenced by an identifier of an existing session map matching the particular GTP packet's identifier (e.g., IMSI), then control passes to block 332. Otherwise, control passes to block 322.

[0055] In block 332, the GCC can generate an analytic server identifier that is based on a combination of the particular GTP packet's source IP address and destination IP address. In some embodiments, the GCC generates the analytic server identifier by inputting the combination into a specified function—the same specified function used in block 310 of FIG. 3A. The GCC can obtain these addresses from the particular GTP-C packet's outer header. As is discussed above in connection with block 310 of FIG. 3A, the specified function can include a hash algorithm. In one embodiment, the GCC can input additional or alternative data into the specified function. In some embodiments, the analytic server identifier indicates a line card's port (e.g., card and port combination) through which duplicate packets can be forwarded out

of the network switch toward the associated analytic server.

[0056] In block 334, the GCC can determine whether the analytic server identifier generated in block 332 is the same as an analytic server identifier (potentially a card-and-port pair through which an analytic server can be reached) that the matching session map associates with its session map entries. In doing so, the GCC determines whether the session map entries in the matching session map contain outdated information. For example, if the analytic server identifier generated in block 332 is not the same as the analytic server identifier in the matching session map's entries, then this implies that some part of the combination input into the function used to generate the analytic server identifier has changed since the session map was constructed (a process for which is described below starting in block 322). The source (e.g., SGSN) IP address specified in the GTP-C packet might differ from the source IP address that was previously detected in a former GTP-C packet at the generation of the session map, for example. The session map entries are discussed in further detail below. GCC 120 can make this determination based on session map entries stored in session map 204, assuming that session map 204 is the matching session map. If the pairs match, then this implies that the mobile device has not moved to a different sub-network, and so the GCC does not need to update the GCL; under such circumstances, the technique illustrated in FIG. 3B concludes relative to the particular GTP-C packet. Alternatively, if the analytic server identifiers from the specified function and the matching session map's entries are not the same, then control passes to block 336. If there was no hit in block 308 of FIG. 3A, then this implies either that the GCC has not yet received any duplicate GTP-C packet from the mobile device, or that the mobile device has moved into a different sub-network, causing the combination of attribute values of the current duplicate GTP-C packet to differ from the combination of attribute values last inserted into the GCL and into the matching session map's entries.

[0057] If the analytic server identifiers do not match, due to a combination of attribute values having changed as discussed above, then this indicates that the mobile device has moved out of range of one cellular telephone tower (or other network entry point) and into the range of another cellular telephone tower (or other network entry point). As a result, the combination of source and destination IP addresses (e.g., for an SGSN and GGSN in the case of a 3G network) indicated in the particular GTP-C packet having a particular IMSI differ from the combination indicated in the session map entries of the session map also specifying that particular IMSI. Typically, in the case of a mismatch, the source IP addresses will differ, while the destination IP addresses will have remained the same. In block 336, the GCC can update the session map entries of the matching session map to reflect the new source and destination IP addresses of the particular GTP-C packet, replacing the existing source and desti-

nation IP addresses in those session map entries—though, as discussed above, the destination IP address might remain the same. However, the analytic server identifier (potentially a card-and-port pair) to which those updated session map entries are mapped remains the same, to ensure that packets belonging to the session identified by the particular IMSI continue to be routed internally toward the same one of analytic servers 130A-N as occurred before the mobile device's movement. Control passes to block 328.

[0058] Alternatively, in block 322, the GCC can create a new session map based on information contained in the particular GTP-C packet. For example, GCC 120 can create and store session map 204. In one embodiment, the session map maps a session identifier, such as an IMSI, that is specified in the particular GTP-C packet, to designated attributes that are also specified in the particular GTP-C packet. For example, GCC 120 can create session map 204 to map the particular GTP-C packet's IMSI to designated attributes such as a source IP address, a destination IP address, a transmission control protocol (TCP) port, and a tunnel identifier (TEID) specified in the particular GTP-C packet.

[0059] In an embodiment, session map 204 associates the GTP-C packet's IMSI with each of four separate session map entries. Two session map entries are for GTP-C packets outbound from a mobile device (GTP-C uplink) and GTP-U packets outbound from the mobile device (GTP-U uplink), respectively. Two session map entries are for GTP-C packets inbound to the mobile device (GTP-C downlink) and GTP-U packets inbound to the mobile device (GTP-U downlink), respectively. Upon receiving an outbound GTP-C packet, the GCC can populate the session map entries for the GTP-C and GTP-U uplinks. Upon receiving an inbound GTP-C packet, the GCC can populate the session map entries for the GTP-C and GTP-U downlinks. Regardless of whether the particular GTP-C packet is inbound or outbound, GCC 120 can determine that the particular GTP-C packet pertains to session map 204 (as opposed to some other session map) due to the particular GTP-C packet specifying the same IMSI (or alternative session identifier) that is stored in session map 204.

[0060] With reference to FIG. 1, in the case of a 3G network, the following facts pertain to the session map entries. The GTP-C uplink entry specifies attributes of GTP-C traffic having a source IP address of an SGSN (e.g., SGSN 106) and a destination IP address of a GGSN (e.g., GGSN 108). The GTP-C downlink entry specifies attributes of GTP-C traffic having a source IP address of a GGSN (e.g., GGSN 108) and a destination address of an SGSN (e.g., SGSN 106). Similarly, the GTP-U uplink entry specifies attributes of GTP-U traffic having a source IP address of an SGSN (e.g., SGSN 106) and a destination IP address of a GGSN (e.g., GGSN 108). The GTP-U downlink entry specifies attributes of GTP-U traffic having a source IP address of a GGSN (e.g., GGSN 108) and a destination address of an SGSN (e.g., SGSN 106).

[0061] The IP addresses of the SGSN and the GGSN are pertinent in the case of a 3G network because the tap or other duplication mechanism that diverts duplicate traffic to network switch 118 is situated in between the SGSN (e.g., SGSN 106) and the GGSN (e.g., GGSN 108). The actual SGSN and GGSN IP addresses placed in the entries of session map 204 may depend on which of several sub-networks (each having a separate SGSN and GGSN) the traffic flowed through, which may in turn depend on which of several cellular telephone towers (or other network entry points) interfaced wirelessly with the mobile device.

[0062] Referring again to FIG. 3, in block 324, the GCC can generate an analytic server identifier based on a combination of the GTP-C packet's attribute values. For example, the GCC can generate the analytic server identifier by inputting a combination of the particular GTP-C packet's source IP address and destination IP address into the specified function—the same specified function used in block 310 of FIG. 3A. The GCC can obtain these addresses from the particular GTP-C packet's outer header. In some embodiments, the GCC can input additional or alternative data into the specified function.

[0063] In block 326, the GCC can map each session map entry in the session map to the analytic server identifier (e.g., a card-and-port pair (e.g., 15/1)) produced in block 324 (e.g., by the specified function). This analytic server identifier can identify the line card and the port to which packets having the entry's designated attributes are to be internally directed within the network switch. For example, each session map entry of a particular session map may initially be mapped to card-and-port pair 15/1, indicating that entry-matching packets are to be internally routed to line card 212 (in slot 15), and more specifically, port 1 of that line card.

[0064] In block 328, the GCC can program the GCL with GCL entries that reflect the information in the session map entries. For example, for each session map entry in session map 204, GCC 120 can program GCL 216 maintained by line card 208 to include a GCL entry that indicates that packets having a specified source IP address, destination IP address, TCP port, and tunnel identifier are to be forwarded to a specified card-and-port pair (e.g., 15/1). The GCL thereby becomes consistent with the session map. Consequently, duplicate packets that are internally routed in response to a GCL hit (per block 314 of FIG. 3A) can be internally routed based on information contained in or derived from the appropriate session map. The technique illustrated in FIG. 3B then concludes relative to the particular GTP-C packet.

MULTIPLE SESSION MAPS FOR MULTIPLE COMMUNICATION SESSIONS

[0065] FIG. 4 is a system diagram illustrating an example of a system 400 including a GCC that stores multiple different session maps for multiple different communication sessions, according to some embodiments. Sys-

tem 400 includes components similar to those included in system 200 of FIG. 2. However, in system 400, GCC 120 stores two different session maps: session map 204 (referenced above), and another session map 402. As is discussed above in connection with blocks 322-326 of FIG. 3B, when the GCC receives a GTP-C packet that contains a session identifier (e.g., an IMSI number) that does not match the session identifier of any existing session map maintained by the GCC, the GCC can responsively create a new session map and stores the GTP-C packet's session identifier in that new session map as that session map's session identifier.

[0066] Thus, for example, if GCC 120 receives a first GTP-C packet that specifies a first IMSI, but determines that no existing session map is associated with the first IMSI (perhaps because GCC 120 does not store any session maps initially), then GCC 120 can create a first session map, such as session map 204, and associate the first session map with the first IMSI. GCC 120 can populate session map entries of the first session map with attributes of the first GTP-C packet, such as the first GTP-C packet's source and destination IP addresses. GCC 120 can map the session map entries of the first session map to a card-and-port pair that are identified by the output of a specified function to which GCC 120 supplied at least some of the first GTP-C packet's attributes as input.

[0067] Continuing the example, if GCC 120 later receives a second GTP-C packet that specifies a second IMSI that differs from the first IMSI, and therefore again determines that no existing session map is associated with the second IMSI, then GCC 120 can create a second session map, such as session map 402, and associate the second session map with the second IMSI. GCC 120 can populate session map entries of the second session map with attributes of the second GTP-C packet, such as the second GTP-C packet's source and destination IP addresses. GCC 120 can map the session map entries of the second session map to a card-and-port pair that are identified by the output of the specified function to which GCC 120 supplies at least some of the second GTP-C packet's attributes as input.

[0068] Continuing the example, if GCC 120 later receives a third GTP-C packet that specifies either the first or second IMSI, then, instead of creating a new session map, GCC 120 can select, from among the existing first and second session maps, a particular session map that is already associated with the third GTP-C packet's IMSI. GCC 120 can determine whether inputting the attributes of the third GTP-C packet into the specified function produces output that identifies the same card-and-port pair with which the session map entries of the particular session map are already associated. If the card-and-port pairs are identical, then the particular session map does not need to be updated.

[0069] However, if the card-and-port pairs are not identical, then GCC 120 can take measures to ensure that the third GTP-C packet's attributes will be mapped to the same card-and-port pair that was originally associated

with the session map entries of the particular session map. In one embodiment, GCC 120 can accomplish this by updating the particular session map's session map entries to contain the third GTP-C packet's revised attributes, but leave unchanged the original card-and-port pair with which those session map entries are already associated.

[0070] After updating any session map in this manner, GCC 120 can program GCL 216 to reflect the updated session map information, so that subsequent GTP-U packets belonging to the same session but containing the third GTP-C packet's attributes (yet no IMSI) will produce a GCL hit that causes those GTP-U packets to be directed to the same one of analytic servers 130A-N as their predecessor GTP-U packets that specified different attributes.

[0071] As a result of the foregoing techniques, in some embodiments, as soon as a particular mobile device becomes associated with a particular one of analytic servers 130A-N via the creation of a session map for that mobile device's IMSI, all of the packets duplicated from traffic flowing to or from that mobile device will be directed to that particular analytic server thereafter. This will be the case even if that mobile device's movement causes its traffic to flow through various different SGSN-GGSN pairs or MME-SGW pairs over time. The mobile device gets "stuck" to the same particular analytic server, so that the particular analytic server has the complete picture of all of the mobile device's traffic when performing analysis relative to that traffic.

EXAMPLE SESSION MAPS

[0072] FIG. 5 is a diagram illustrating an example of a populated session map 500, according to some embodiments. Session map 500 includes an IMSI number 502, a GTP-C uplink session map entry 504, a GTP-C downlink session map entry 506, a GTP-U uplink session map entry 508, a GTP-U downlink session map entry 510, a GTP-U uplink session map entry 511, and a GTP-U downlink session map entry 512. Each of session map entries 504-512 includes various attributes, such as source IP address 513, a destination IP address 514, a TCP port 516, and a tunnel identifier 518.

[0073] The source IP addresses for uplink session map entries 504 and 506 will typically match the destination IP addresses for downlink session map entries 508 and 510. Conversely, the destination IP addresses for uplink session map entries 504 and 506 will typically match the source IP addresses for downlink session map entries 508 and 510.

[0074] In situations where the GTP-C packet arrived at network switch 118 through a 3G network, the uplink source IP addresses and downlink destination IP addresses will typically be an SGSN's IP address, and the the downlink source IP addresses and uplink destination IP addresses will typically be an GGSN's IP address. These will be the IP addresses of the SGSN and GGSN

on either end of the tapped connection, if a tap is used as a duplication mechanism.

[0075] Each of session map entries 504-510 is mapped to a card-and-port pair 520 that can be determined from the output of a specified function into which attributes (e.g., source and destination IP address combination) of the initial GTP-C packet that caused GCC 120 to create session map 500 were input. As is discussed above, in an embodiment, these card-and-port pairs 520 mapped to session map entries 504-510 remain constant, even if subsequent GTP-C packets bearing the same IMSI number 502 cause any or all of attributes 513-518 to change.

[0076] FIG. 9 is a diagram illustrating another example of a populated session map 900, according to some embodiments. Session map 900 includes an IMSI number 902, a GTP-C uplink session map entry 904, a GTP-C downlink session map entry 906, a GTP-U uplink session map entry 908, a GTP-U downlink session map entry 910, a GTP-U uplink session map entry 911, and a GTP-U downlink session map entry 912. Each of session map entries 904-910 includes various attributes, such as source IP address 913, a destination IP address 914, a TCP port 916, and a tunnel identifier 918.

[0077] The source IP addresses for uplink session map entries 904 and 906 will typically match the destination IP addresses for downlink session map entries 908 and 910. Conversely, the destination IP addresses for uplink session map entries 904 and 906 will typically match the source IP addresses for downlink session map entries 908 and 910.

[0078] In situations where the GTP-C packet arrived at network switch 118 through a 4G/LTE network, the uplink source IP addresses and downlink destination IP addresses will typically be an MME's IP address, and the the downlink source IP addresses and uplink destination IP addresses will typically be an SGW's IP address. These will be the IP addresses of the MME and SGW on either end of the tapped connection, if a tap is used as a duplication mechanism.

[0079] Each of session map entries 904-910 is mapped to a card-and-port pair 920 that can be determined from the output of a specified function into which attributes (e.g., source and destination IP address combination) of the initial GTP-C packet that caused GCC 120 to create session map 900 were input. As is discussed above, in an embodiment, these card-and-port pairs 920 mapped to session map entries 904-910 remain constant, even if subsequent GTP-C packets bearing the same IMSI number 902 cause any or all of attributes 913-918 to change.

NETWORK SWITCH CONTAINING GCC AND ANALYTIC MODULES

[0080] FIG. 8 is a system diagram illustrating an example of a network switch that contains a GCC and analytic modules, according to some embodiments. Al-

though some embodiments described above involve a network switch interacting with a GCC and analytic servers that may be external to the network switch, the network switch illustrated in FIG. 8 contains both the GCC and analytic modules that perform the same functions performed by the analytic servers described above.

[0081] System 800 includes components similar to those illustrated in system 400 of FIG. 4. However, in system 800, network switch 818 includes GCC 820 and analytic modules 830A-N. Analytic modules 830A-N perform functions described above as being performed by analytic servers. In an embodiment illustrated in FIG. 8, network switch 818 does not need to interface with any external analytic servers.

[0082] Analytic modules 830A-N can be processes that execute in parallel within network switch 818. These processes can execute on hardware contained within a line card, which can be modified specifically to store and to execute these processes. Analytic modules 830A-N can store the results of their analytic processing relative to duplicate packets within storage memories contained within such a line card. Additionally or alternatively, analytic modules 830A-830N can execute in parallel on one or more master central processing units or on one or more cores of such central processing units contained within network switch 818. Additionally or alternatively, analytic modules 830A-N can execute within a virtual machine executing within network switch 818 on any of the hardware components described above.

[0083] Analytic modules 830A-N can perform traffic analysis relative to data packets flowing through the network. Such packets can be duplicate packets that have been diverted to analytic modules 830A-N. In some embodiments, all traffic belonging to a particular communication session is diverted to the same analytic module under circumstances in which multiple analytic modules are analyzing the network traffic. Each of analytic modules 830A-N can receive packet copies, classify those packet copies into different types based on rules and packet attribute values, and determine network sources from which different types of traffic originate. In this manner, for example, network traffic that appears to be associated with an attack, such as a denial-of-service attack, can be identified, and the sources of the attack can be ascertained.

[0084] GCC 820 can automatically program network switch 818 to forward copies of packets originating from a mobile device and having a shared attribute to the same one of analytic modules 830A-N. Network switch 818 thereby sends the packet copies having the shared attribute to the same analytic module, regardless of the regions into which the mobile device moves.

[0085] In response to detecting a changed source attribute (e.g., SGSN) within a control packet originating from a mobile device, GCC 820 can update a session map specific to that mobile device in order to cause packets having that changed source attribute to be forwarded to the same one of analytic modules 830A-N to which

packets having the former source attribute were being forwarded prior to the change. As a result, network switch 818 can ensure that packets belonging to a particular session still are forwarded to the same analytic module even if the mobile device has moved to a different region.

[0086] Instead of calculating an analytic server identifier as described above, a GTP services card (e.g., card 208) can compute an analytic module identifier. Each of analytic modules 830AN can be associated with a separate and unique analytic module identifier. Such an identifier may, but does not need to, specify a card-and-port pair. In some embodiments, the GTP services card includes multiple ports, each of which interfaces with a separate one of analytic modules 830A-N. In such embodiments, the GTP services card can forward duplicate packets through a port that is associated with the one of analytic modules 830A-N selected using techniques similar to those described above. GCL 216 and session maps 202 and 402 can store and use analytic module identifiers in place of the analytic server identifiers discussed above.

EXAMPLE NETWORK NODE

[0087] Various different systems and devices may incorporate an embodiment of the present invention. FIG. 6 provides an example of a network device that may incorporate an embodiment of the present invention. FIG. 6 depicts a simplified block diagram of a network device 600 that may incorporate an embodiment of the present invention (e.g., network device 600 may correspond to nodes depicted in figures above). In the embodiment depicted in FIG. 6, network device 600 comprises a plurality of ports 612 for receiving and forwarding data packets and multiple cards that are configured to perform processing to facilitate forwarding of the data packets to their intended destinations. The multiple cards may include one or more line cards 604 and a management card 602. In one embodiment, a card, sometimes also referred to as a blade or module, can be inserted into one of a plurality of slots on the chassis of network device 600. This modular design allows for flexible configurations with different combinations of cards in the various slots of the device according to differing network topologies and switching requirements. The components of network device 600 depicted in FIG. 6 are meant for illustrative purposes only and are not intended to limit the scope of the invention in any manner. Alternative embodiments may have more or less components than those shown in FIG. 6.

[0088] Ports 612 represent the I/O plane for network device 600. Network device 600 is configured to receive and forward packets using ports 612. A port within ports 612 may be classified as an input port or an output port depending upon whether network device 600 receives or transmits a data packet using the port. A port over which a packet is received by network device 600 is referred to as an input port. A port used for communicating or forwarding a packet from network device 600 is re-

ferred to as an output port. A particular port may function both as an input port and an output port. A port may be connected by a link or interface to a neighboring network device or network. Ports 612 may be capable of receiving and/or transmitting different types of traffic at different speeds including 1 Gigabit/sec, 6 Gigabits/sec, 60 Gigabits/sec, or even more. In some embodiments, multiple ports of network device 600 may be logically grouped into one or more trunks.

[0089] Upon receiving a data packet via an input port, network device 600 is configured to determine an output port of device 600 to be used for transmitting the data packet from network device 600 to facilitate communication of the packet to its intended destination. Within network device 600, the packet is forwarded from the input port to the determined output port and then transmitted from network device 600 using the output port. In one embodiment, forwarding of packets from an input port to an output port is performed by one or more line cards 604. Line cards 604 represent the data forwarding plane of network device 600. Each line card may comprise one or more packet processors that are programmed to perform forwarding of data packets from an input port to an output port. In one embodiment, processing performed by a line card may comprise extracting information from a received packet, performing lookups using the extracted information to determine an output port for the packet such that the packet can be forwarded to its intended destination, and to forward the packet to the output port. The extracted information may include, for example, the header of the received packet.

[0090] Management card 602 is configured to perform management and control functions for network device 600 and represents the management plane for network device 600. In one embodiment, management card 602 is communicatively coupled to line cards 604 via switch fabric 606. Management card 602 may comprise one or more physical processors 608, one or more of which may be multicore processors. These management card processors may be general purpose multicore microprocessors such as ones provided by Intel, AMD, ARM, Freescale Semiconductor, Inc., and the like, that operate under the control of software stored in associated memory 610. The processors may run one or more VMs. Resources allocated to these VMs may be dynamically changed. In some embodiments, multiple management cards may be provided for redundancy and to increase availability.

[0091] In some embodiments, one or more line cards 604 may each comprise one or more physical processors 614, some of which may be multicore. These processors may run one or more VMs. Resources allocated to these VMs may be dynamically changed.

[0092] The embodiment depicted in FIG. 6 depicts a chassis-based system. This however is not intended to be limiting. Certain embodiments of the present invention may also be embodied in non-chassis based network devices, which are sometimes referred to as "pizza boxes." Such a network device may comprise a single phys-

ical multicore CPU or multiple physical multicore CPUs.

[0093] Various embodiments described above can be realized using any combination of dedicated components and/or programmable processors and/or other programmable devices. The various embodiments may be implemented only in hardware, or only in software, or using combinations thereof. For example, the software may be in the form of instructions, programs, etc. stored in a computer-readable memory and may be executed by one or more processing units, where the processing unit is a processor, a core of a processor, or a percentage of a core. In certain embodiments, the various processing described above, including the processing depicted in the flowcharts described above can be performed in software without needing changes to existing device hardware (e.g., router hardware), thereby increasing the economic viability of the solution. Since certain inventive embodiments can be implemented entirely in software, it allows for quick rollouts or turnarounds along with lesser capital investment, which further increases the economic viability and attractiveness of the solution.

[0094] The various processes described herein can be implemented on the same processor or different processors in any combination, with each processor having one or more cores. Accordingly, where components or modules are described as being adapted to or configured to perform a certain operation, such configuration can be accomplished, e.g., by designing electronic circuits to perform the operation, by programming programmable electronic circuits (such as microprocessors) to perform the operation, by providing software or code instructions that are executable by the component or module (e.g., one or more processors) to perform the operation, or any combination thereof. Processes can communicate using a variety of techniques including but not limited to conventional techniques for interprocess communication, and different pairs of processes may use different techniques, or the same pair of processes may use different techniques at different times. Further, while the embodiments described above may make reference to specific hardware and software components, those skilled in the art will appreciate that different combinations of hardware and/or software components may also be used and that particular operations described as being implemented in hardware might also be implemented in software or vice versa.

[0095] The various embodiments are not restricted to operation within certain specific data processing environments, but are free to operate within a plurality of data processing environments. Additionally, although embodiments have been described using a particular series of transactions, this is not intended to be limiting.

[0096] Thus, although specific invention embodiments have been described, these are not intended to be limiting. The scope of the invention is defined by the appended claims.

Claims

1. A network node (118, 600) configured to perform operations comprising:
 - receiving (702) a first packet from a first network entry point;
 - receiving (704) a second packet from a second network entry point that differs from the first network entry point;
 - determining (706) that the second packet belongs to a same session to which the first packet belongs;
 - determining that the second packet contains a second combination of attributes that differs from a first combination of attributes contained in the first packet;
 - in response to determining that the second combination of attributes differs from the first combination of attributes, replacing a mapping between the first combination of attributes and an identifier of an analytic server (130A-130N) to which the first packet was forwarded with a mapping between the second combination of attributes and the identifier of the analytic server (130A-130N); and
 - after the replacing, forwarding (708) packets that contain the second combination of attributes to the analytic server (130A-130N) having the identifier that is mapped to the second combination of attributes.
2. The network node (118, 600) of claim 1, wherein the first and second packets are GTP-C packets.
3. The network node (118, 600) of claim 1, wherein the replacing comprises:
 - updating a mapping that is associated with an IMSI of a mobile device from which the first packet originated.
4. The network node (118, 600) of claim 1, wherein routing the packets that contain the second combination of attributes comprises routing GTP-U packets that specify the second set of attributes.
5. The network node (118, 600) of claim 1, wherein a set of attributes to which the identifier of the analytic server (130A-130N) is mapped includes a source IP address, a destination IP address, a TCP port, and a tunnel identifier.
6. The network node (118, 600) of claim 5, wherein the source IP address is an SGSN IP address, and wherein the destination IP address is a GGSN IP address.
7. The network node (118, 600) of claim 5, wherein the

source IP address is an MME IP address, and where-
in the destination IP address is a SGW IP address.

8. The network node (118, 600) of claim 1, wherein replacing the mapping between the first combination of attributes and the identifier of the analytic server (130A-130N) with the mapping between the second combination of attributes and the identifier of the analytic server (130A-130N) comprises:

inputting at least a subset of the second combination of attributes into a specified function; and determining a card and port pair based on the output of the specified function, the card and port pair identifying a line card (208, 212, 604) and a port (612) of the network node (118, 600) that are communicatively coupled with the analytic server (130A-130N).

9. The network node (118, 600) of claim 8, wherein routing packets that contain the second combination of attributes to the analytic server (130A-130N) having the identifier that is mapped to the second combination of attributes comprises:

determining whether an access control list contains an entry that contains the second combination of attributes; and responsive to determining that the access control list does not contain any entry that contains the second combination of attributes, determining the card and port pair based on the output of the specified function.

10. The network node (118, 600) of claim 1, further comprising:
responsive to the replacing, altering an access control list to reflect changes produced by the replacing.

11. The network node (118, 600) of claim 1, wherein:

the first network entry point is a first cellular telephone tower with which a mobile device interfaces when the mobile device originates the first packet; and
the second network entry point is a second cellular telephone tower with which the mobile device interfaces when the mobile device originates the second packet.

12. The network node (118, 600) of claim 1, wherein:

the first packet specifies a first source IP address and a particular destination IP address;
the second packet specifies a second source IP address and the particular destination IP address; and
the first source IP address differs from the sec-

ond source IP address.

13. The network node (118, 600) of claim 1, wherein:

the first packet specifies a first SGSN and a particular GGSN;
the second packet specifies a second SGSN and the particular GGSN; and
the first SGSN differs from the second SGSN.

14. The network node (118, 600) of claim 1, wherein:

the first packet specifies an IMSI; and
the second packet does not specify the IMSI.

Patentansprüche

1. Netzwerkknoten (118, 600), der konfiguriert ist, Operationen durchzuführen, die umfassen:

Empfangen (702) eines ersten Pakets von einem ersten Netzwerkeintrittspunkt;
Empfangen (704) eines zweiten Pakets von einem zweiten Netzwerkeintrittspunkt, der sich von dem ersten Netzwerkeintrittspunkt unterscheidet;

Bestimmen (706), dass das zweite Paket zu derselben Sitzung gehört, zu der das erste Paket gehört;

Bestimmen, dass das zweite Paket eine zweite Kombination von Attributen enthält, die sich von einer ersten Kombination von Attributen unterscheidet, die in dem ersten Paket enthalten sind; als Antwort auf das Bestimmen, dass sich die zweite Kombination von Attributen von der ersten Kombination von Attributen unterscheidet, Ersetzen eines Mappings zwischen der ersten Kombination von Attributen und einer Kennung eines analytischen Servers (130A-130N), an den das erste Paket weitergeleitet wurde, mit einem Mapping zwischen die zweite Kombination von Attributen und der Kennung des analytischen Servers (130A-130N); und nach dem Ersetzen, Weiterleiten (708) von Paketen, welche die zweite Kombination von Attributen enthalten, an den analytischen Server (130A-130N), welcher die Kennung aufweist, die auf die zweite Kombination von Attributen gemappt ist.

2. Netzwerkknoten (118, 600) nach Anspruch 1, wobei das erste und das zweite Paket GTP-C-Pakete sind.

3. Netzwerkknoten (118, 600) nach Anspruch 1, wobei das Ersetzen umfasst:

Aktualisieren eines Mappings, das mit einer IMSI einer mobilen Vorrichtung verknüpft ist, von der das

erste Paket stammt.

4. Netzwerkknoten (118, 600) nach Anspruch 1, wobei das Routen der Pakete, welche die zweite Kombination von Attributen enthalten, das Weiterleiten von GTP-U-Paketen umfasst, die den zweiten Satz von Attributen spezifizieren.

5. Netzwerkknoten (118, 600) nach Anspruch 1, wobei ein Satz von Attributen, auf welche die Kennung des analytischen Servers (130A - 130N) gemappt ist, eine Quell-IP-Adresse, eine Ziel-IP-Adresse, einen TCP-Port und eine Tunnelkennung enthält.

6. Netzwerkknoten (118, 600) nach Anspruch 5, wobei die Quell-IP-Adresse eine SGSN-IP-Adresse ist und wobei die Ziel-IP-Adresse eine GGSN-IP-Adresse ist.

7. Netzwerkknoten (118, 600) nach Anspruch 5, wobei die Quell-IP-Adresse eine MME-IP-Adresse ist und wobei die Ziel-IP-Adresse eine SGW-IP-Adresse ist.

8. Netzwerkknoten (118, 600) nach Anspruch 1, wobei das Ersetzen des Mappings zwischen der ersten Kombination von Attributen und der Kennung des analytischen Servers (130A-130N) durch das Mapping zwischen der zweiten Kombination von Attributen und der Kennung des der analytische Server (130A-130N) umfasst:

Eingeben mindestens einer Untergruppe der zweiten Kombination von Attributen in eine spezifizierte Funktion; und
Bestimmen eines Card- und Port-Paares basierend auf der Ausgabe der spezifizierten Funktion, wobei das Card- und Port-Paar eine Line-card (208, 212, 604) und einen Port (612) des Netzwerkknotens (118, 600) identifiziert, die kommunikativ mit dem analytischen Server (130A-130N) gekoppelt sind.

9. Netzwerkknoten (118, 600) nach Anspruch 8, wobei das Routen von Paketen, welche die zweite Kombination von Attributen enthalten, zu dem analytischen Server (130A-130N), welcher die Kennung aufweist, die auf die zweite Kombination von Attributen gemappt ist, umfasst:

Bestimmen, ob eine Zugriffskontrollliste einen Eintrag enthält, der die zweite Kombination von Attributen enthält; und
als Antwort auf das Bestimmen, dass die Zugriffskontrollliste keinen Eintrag enthält, der die zweite Kombination von Attributen enthält, Bestimmen des Card- und Port-Paares basierend auf der Ausgabe der spezifizierten Funktion.

10. Netzwerkknoten (118, 600) nach Anspruch 1, ferner umfassend:

als Antwort auf das Ersetzen, Ändern einer Zugriffskontrollliste, um durch das Ersetzen erzeugte Änderungen widerzuspiegeln.

11. Netzwerkknoten (118, 600) nach Anspruch 1, wobei:

der erste Netzwerkeintrittspunkt ein erster Mobiltelefon- bzw. -funkmast ist, mit dem sich eine mobile Vorrichtung schnittstellenmäßig verbindet, wenn die mobile Vorrichtung das erste Paket erzeugt; und

der zweite Netzwerkeintrittspunkt ein zweiter Mobiltelefon- bzw. -funkmast ist, mit dem sich die mobile Vorrichtung schnittstellenmäßig verbindet, wenn die mobile Vorrichtung das erste Paket erzeugt.

12. Netzwerkknoten (118, 600) nach Anspruch 1, wobei:

das erste Paket eine erste Quell-IP-Adresse und eine bestimmte Ziel-IP-Adresse spezifiziert;
das zweite Paket eine zweite Quell-IP-Adresse und die bestimmte Ziel-IP-Adresse spezifiziert; und
sich die erste Quell-IP-Adresse von der zweiten Quell-IP-Adresse unterscheidet.

13. Netzwerkknoten (118, 600) nach Anspruch 1, wobei:

das erste Paket eine erste SGSN und eine bestimmte GGSN spezifiziert;
das zweite Paket eine zweite SGSN und die bestimmte GGSN spezifiziert; und
sich die erste SGSN von der zweiten SGSN unterscheidet.

14. Netzwerkknoten (118, 600) nach Anspruch 1, wobei:

das erste Paket eine IMSI spezifiziert; und
das zweite Paket die IMSI nicht spezifiziert.

Revendications

1. Noeud de réseau (118, 600) configuré pour réaliser des opérations, comportant :

la réception (702) d'un premier paquet à partir d'un premier point d'entrée dans le réseau ;
la réception (704) d'un second paquet à partir d'un second point d'entrée dans le réseau qui diffère du premier point d'entrée dans le réseau ;
la détermination (706) que le second paquet appartient à la session à laquelle appartient le premier paquet ;
la détermination que le second paquet contient

- une seconde combinaison d'attributs qui diffère d'une première combinaison d'attributs contenue dans le premier paquet ;
 en réponse à la détermination que la seconde combinaison d'attributs diffère de la première
 5 combinaison d'attributs, le remplacement d'un mappage entre la première combinaison d'attributs et un identifiant d'un serveur d'analyse (130A-130N) auquel le premier paquet a été transféré par un mappage entre la seconde
 10 combinaison d'attributs et l'identifiant du serveur d'analyse (130A-130N) ; et
 après le remplacement, le transfert (708) de paquets qui contiennent la seconde combinaison d'attributs au serveur d'analyse (130A-130N) ayant l'identifiant qui est mappé sur la seconde
 15 combinaison d'attributs.
2. Noeud de réseau (118, 600) selon la revendication 1, dans lequel les premier et second paquets sont des paquets GTP-C.
 20
3. Noeud de réseau (118, 600) selon la revendication 1, dans lequel le remplacement comporte :
 la mise à jour d'un mappage qui est associé à un IMSI d'un dispositif mobile duquel provient le premier
 25 paquet.
4. Noeud de réseau (118, 600) selon la revendication 1, dans lequel le routage des paquets qui contiennent la seconde combinaison d'attributs comporte le routage de paquets GTP-U qui spécifient le second jeu d'attributs.
 30
5. Noeud de réseau (118, 600) selon la revendication 1, dans lequel un jeu d'attributs, sur lequel est mappé l'identifiant du serveur d'analyse (130A-130N), inclut une adresse IP source, une adresse IP de destination, un port TCP et un identifiant de tunnel.
 35
 40
6. Noeud de réseau (118, 600) selon la revendication 5, dans lequel l'adresse IP source est une adresse IP SGSN et dans lequel l'adresse IP de destination est une adresse IP GGSN.
 45
7. Noeud de réseau (118, 600) selon la revendication 5, dans lequel l'adresse IP source est une adresse IP MME et dans lequel l'adresse IP de destination est une adresse IP SGW.
 50
8. Noeud de réseau (118, 600) selon la revendication 1, dans lequel le remplacement du mappage entre la première combinaison d'attributs et l'identifiant du serveur d'analyse (130A-130N) par le mappage entre la seconde combinaison d'attributs et l'identifiant du serveur d'analyse (130A-130N) comporte :
 55 l'entrée d'au moins un sous-ensemble de la se-
- conde combinaison d'attributs dans une fonction spécifiée ; et
 la détermination d'une paire carte-port sur la base de la sortie de la fonction spécifiée, la paire
 carte-port identifiant une carte de ligne (208, 212, 604) et un port (612) du noeud de réseau (118, 600) qui sont accouplés en communication avec le serveur d'analyse (130A-130N).
9. Noeud de réseau (118, 600) selon la revendication 8, dans lequel le routage des paquets qui contiennent la seconde combinaison d'attributs vers le serveur d'analyse (130A-130N) ayant l'identifiant qui est mappé sur la seconde combinaison d'attributs comporte :
 la détermination pour savoir si une liste de contrôle d'accès contient une entrée qui contient la seconde combinaison d'attributs ; et
 en réponse à la détermination que la liste de contrôle d'accès ne contient pas d'entrée qui contient la seconde combinaison d'attributs, la détermination de la paire carte-port sur la base de la sortie de la fonction spécifiée.
10. Noeud de réseau (118, 600) selon la revendication 1, comportant en outre :
 en réponse au remplacement, la modification d'une liste de contrôle d'accès afin de refléter les changements provoqués par le remplacement.
11. Noeud de réseau (118, 600) selon la revendication 1, dans lequel :
 le premier point d'entrée dans le réseau est une première tour de téléphonie cellulaire avec laquelle un dispositif mobile s'interface lorsque le dispositif mobile provient du premier paquet ; et
 le second point d'entrée dans le réseau est une seconde tour de téléphonie cellulaire avec laquelle le dispositif mobile s'interface lorsque le dispositif mobile provient du second paquet.
12. Noeud de réseau (118, 600) selon la revendication 1, dans lequel :
 le premier paquet spécifie une première adresse IP source et une adresse IP de destination particulière ;
 le second paquet spécifie une seconde adresse IP source et l'adresse IP de détermination particulière ; et
 la première adresse IP source diffère de la seconde adresse IP source.
13. Noeud de réseau (118, 600) selon la revendication 1, dans lequel :

le premier paquet spécifie un premier SGSN et
un GGSN particulier ;
le second paquet spécifie un second SGSN et
le GGSN particulier ; et
le premier SGSN diffère du second SGSN.

5

14. Noeud de réseau (118, 600) selon la revendication
1, dans lequel :

le premier paquet spécifie un IMSI ; et
le second paquet ne spécifie pas l'IMSI.

10

15

20

25

30

35

40

45

50

55

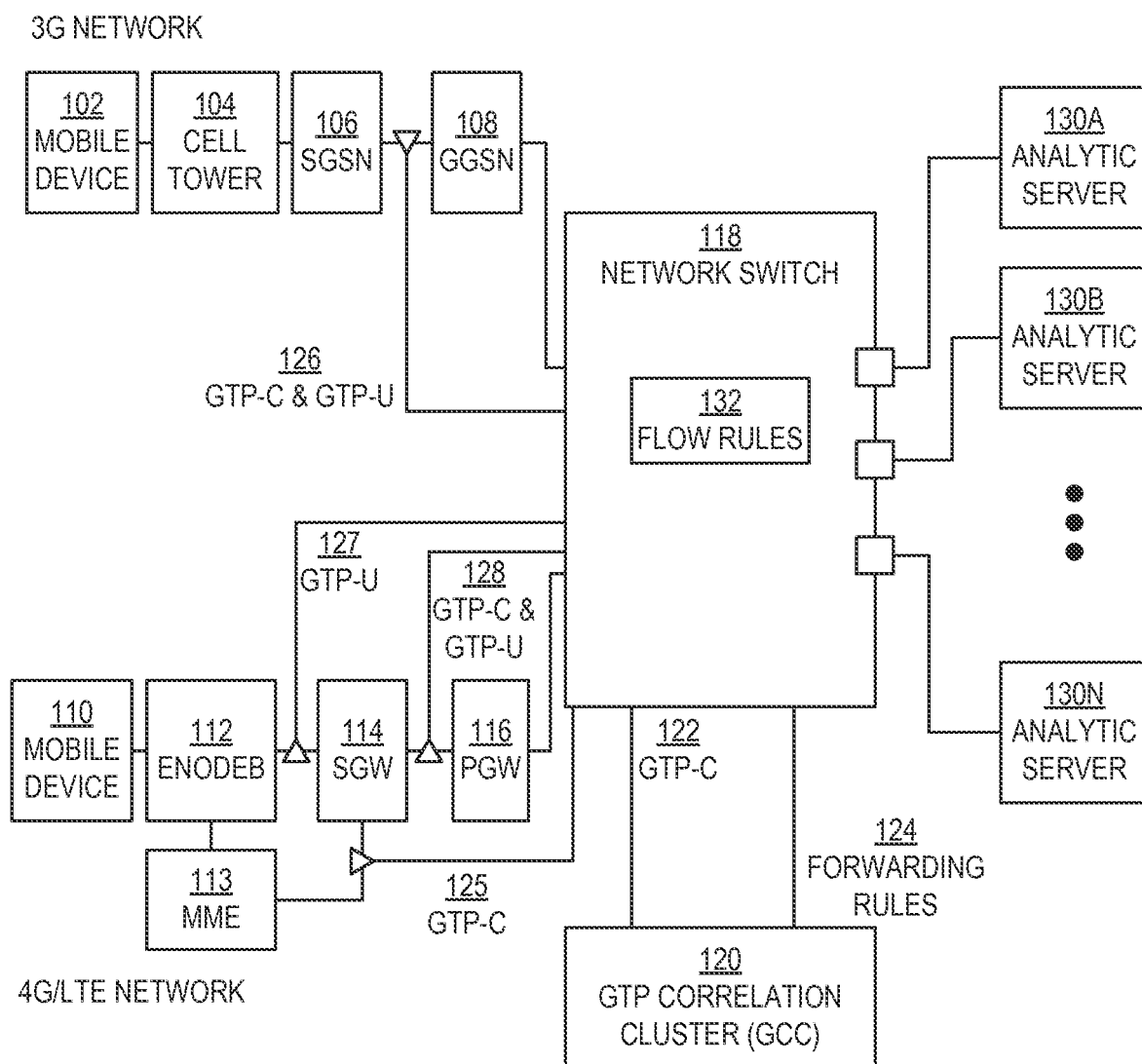


FIG. 1

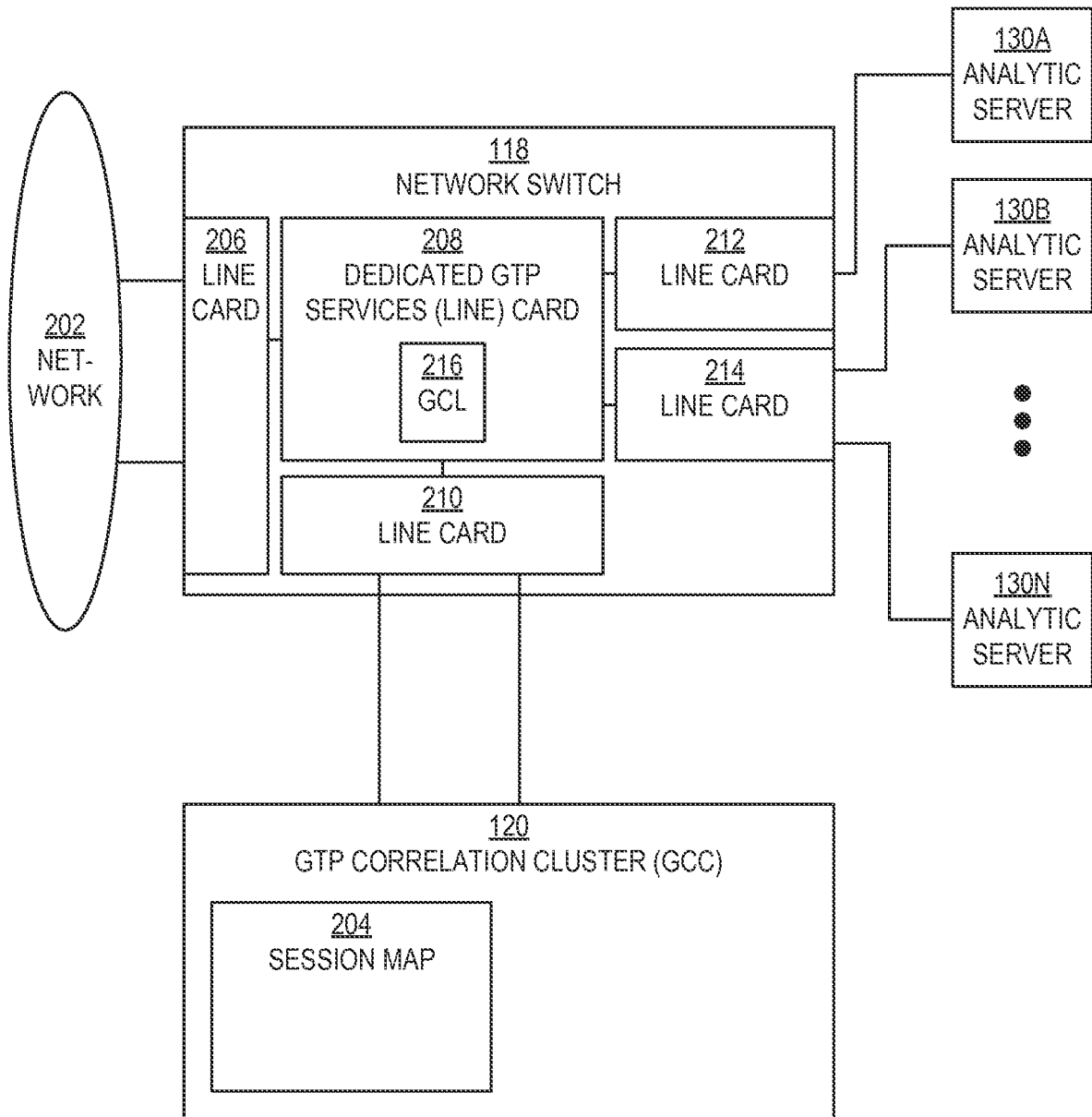


FIG. 2

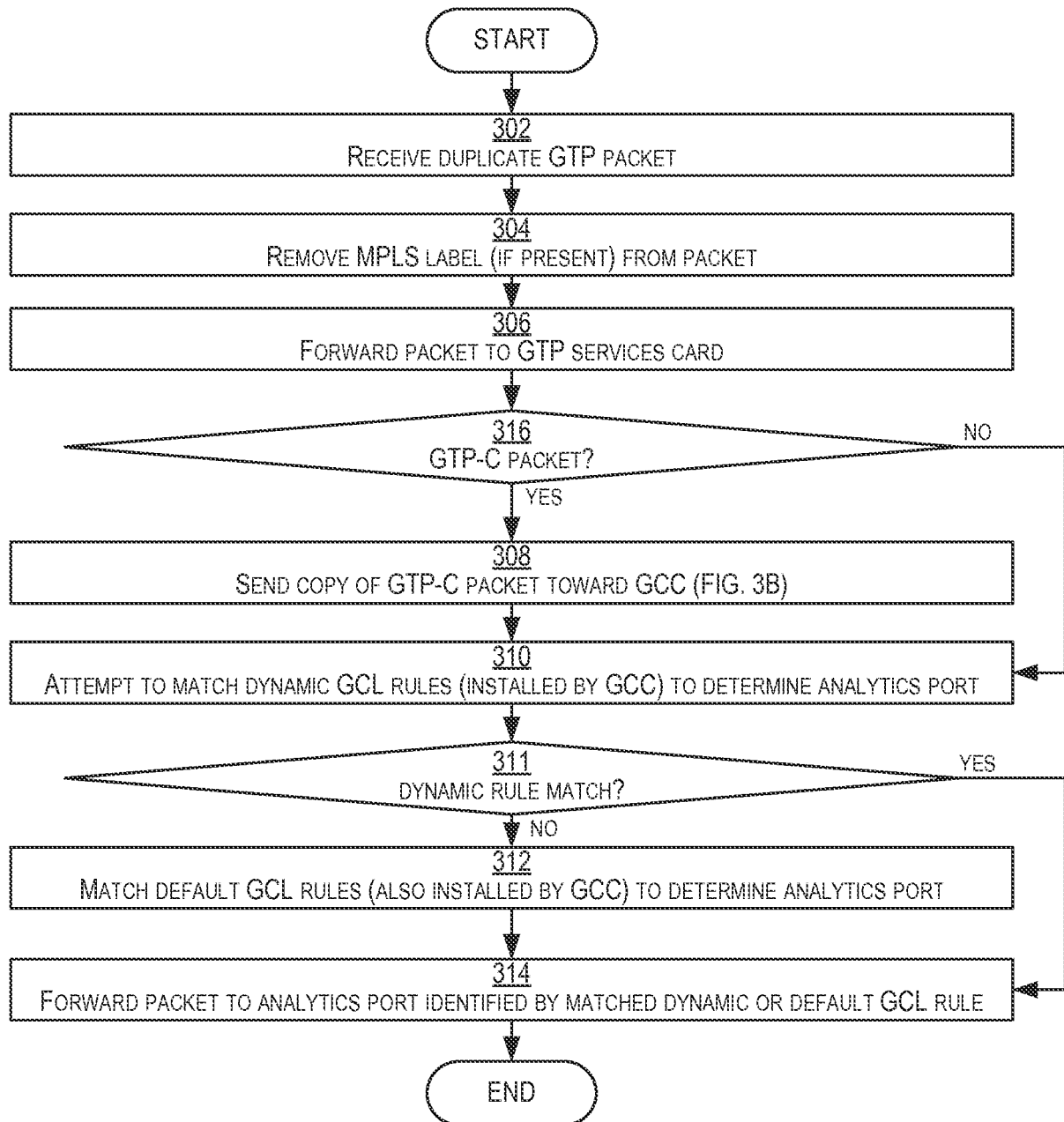


FIG. 3A

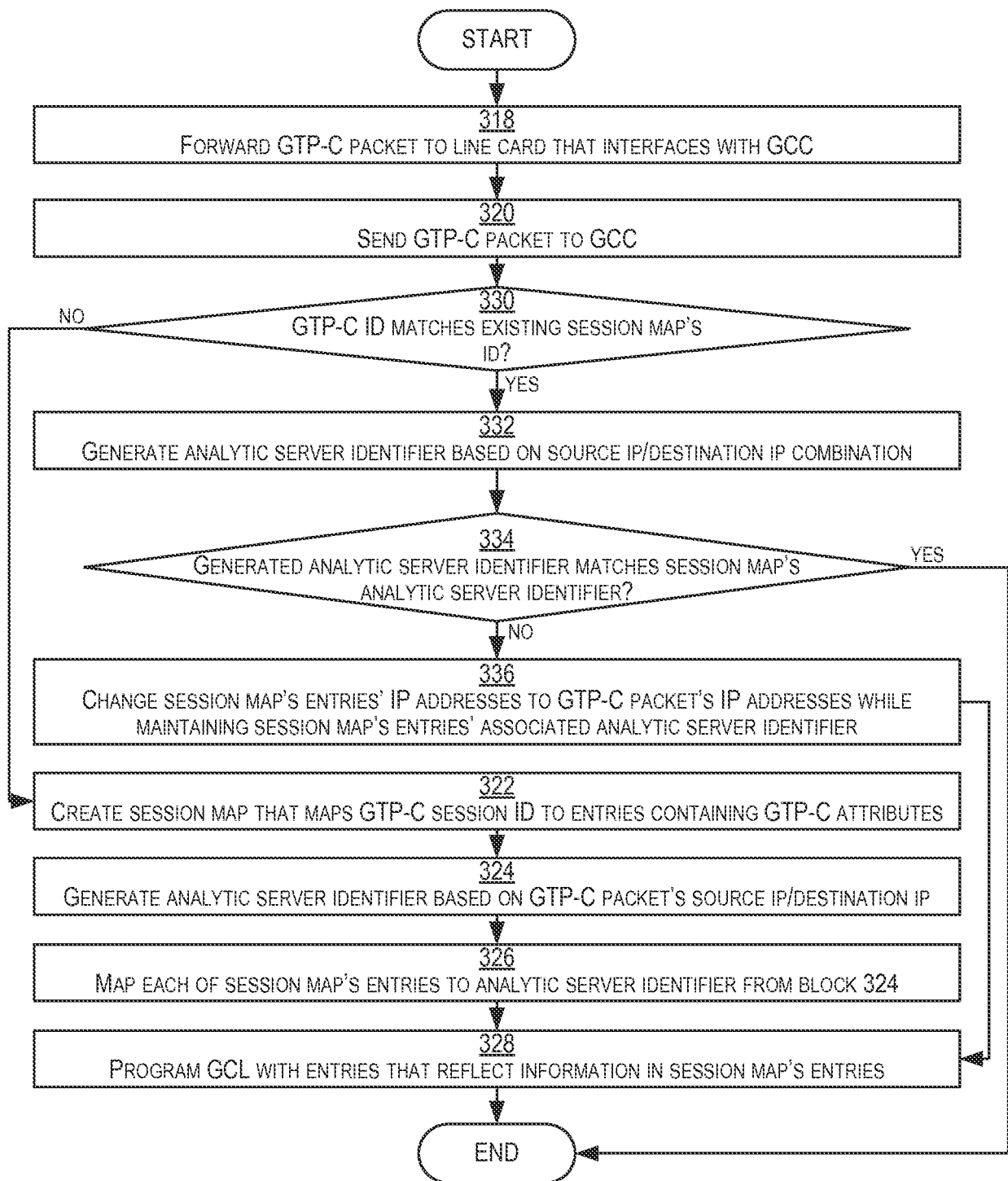


FIG. 3B

400

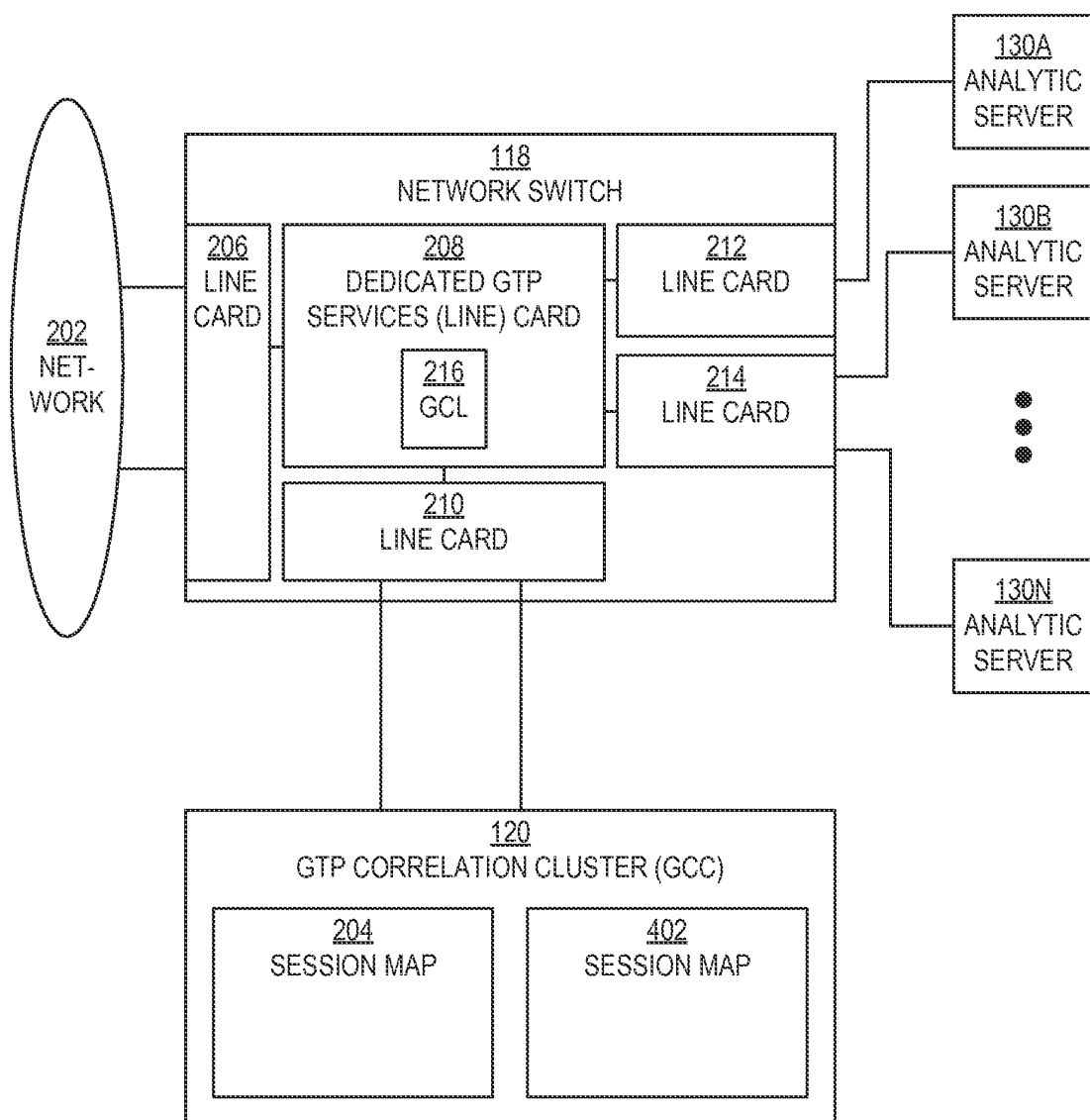


FIG. 4

500

502 IMSI					
	513 SOURCE	514 DESTINATION	516 PORT	518 TUNNEL ID	520 EGRESS
504	SGSN1 IP	GGSN1 IP	PORT C	TEID CU	15/1
506	GGSN1 IP	SGSN1 IP	PORT C	TEID CD	15/1
508	SGSN1 IP	GGSN1 IP	PORT U	TEID U1U	15/1
510	GGSN1 IP	SGSN1 IP	PORT U	TEID U1D	15/1
511	SGSN1 IP	GGSN1 IP	PORT U	TEID U2U	15/1
512	GGSN1 IP	SGSN1 IP	PORT U	TEID U2D	15/1

FIG. 5

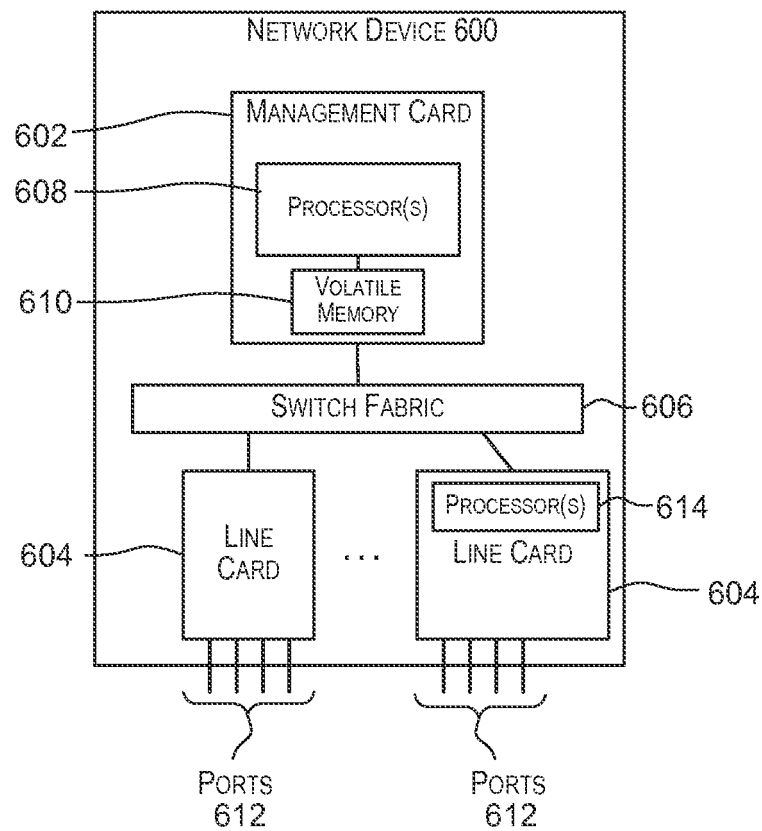


FIG. 6

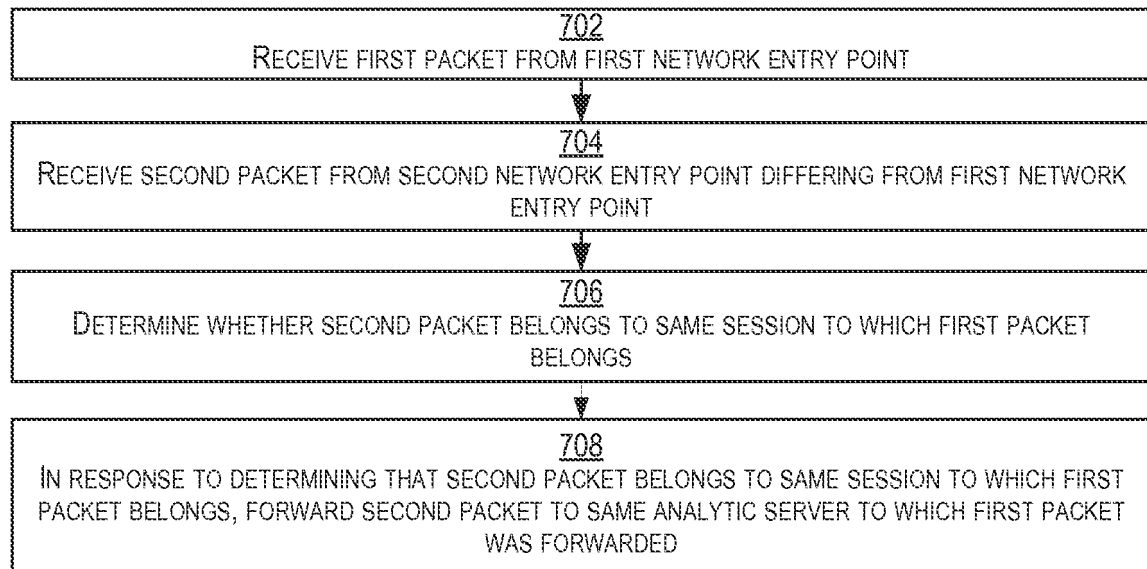


FIG. 7

800

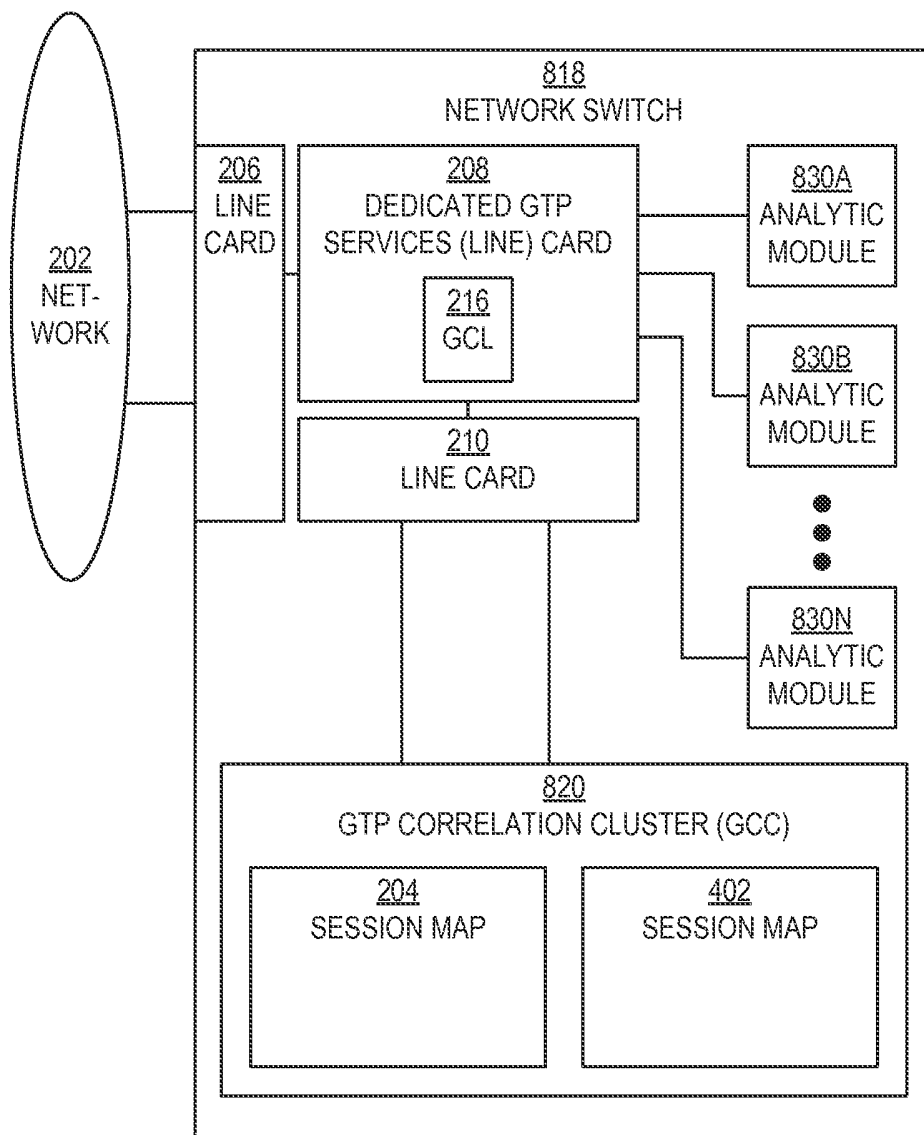


FIG. 8

900

902 IMSI					
913 SOURCE	914 DESTINATION	916 PORT	918 TUNNEL ID	920 EGRESS	
904 MME IP	SGW IP	PORT C	TEID CU	15/1	
906 SGW IP	MME IP	PORT C	TEID CD	15/1	
908 MME IP	SGW IP	PORT U	TEID U1U	15/1	
910 SGW IP	MME IP	PORT U	TEID U1D	15/1	
911 MME IP	SGW IP	PORT U	TEID U2U	15/1	
912 SGW IP	MME IP	PORT U	TEID U2D	15/1	

FIG. 9

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- IE 20070438 [0007]